

INSTRUCTION MANUAL  
MDI-1  
5 or 8 Inch Floppy DISK DRIVER INTERFACE  
FOR  
6800 and 6809 Systems

F & D ASSOCIATES  
1210 Todd Rd  
New Plymouth, Ohio 45654

## TERMS OF SALE

Our boards are sold on an "as is" basis with no expressed or implied warranty of any kind.

## RETURN PRIVILEGE

Our boards and documentation are designed with the advanced hobbyist in mind. If after examining the board and documentation you feel that it is beyond your ability or you don't have proper equipment for checkout, you may return it immediately for a refund of purchase price. You may also return the board if you find the workmanship or quality is not up to your expectations. This return privilege is valid up to 60 days after our shipping date.

There are no warranties of fitness or merchantability, either expressed or implied. In no case are we liable for more than the purchase price of the board.

F & D ASSOCIATES  
1210 Todd Road  
New Plymouth, Ohio 45654

This documentation package is new and , due to human failibility, could contain errors or , more likely, ommissions of information that you may need. We have tried to make it as complete as practical but what is perfectly clear to us may not be so clear to you. We would appreciate you taking the time to let us know of any errors or ommissions. We will do our best to answer any questions you may have.

Be sure to look through the documentation package for any addendums or users notes that may have been added at the last minute. We keep records of board purchasers and will send along any pertinent updates if any occur.

## MDI-1 USERS MANUAL

### MDI-1 Description:

The MDI-1 is a disk controller board for 6800 and 6809 systems using the S50 bus. It plugs into one of the 30 pin I/O slots. The board uses either the WD1771 or the WD1791 Floppy Disk Controller chips. When the 1771 is used, single density 5" and single density 8" drives are supported. When the 1791 is used, single and double density 5" and single density 8" drives are supported. Both 5" and 8" drives can be plugged into the controller at the same time. The board supports up to 4 drives. It is ideal for use with Shugart SA400, Wangco model 82, Shugart SA800/801, Seimens FDD120-8, and other 5" and 8" drives. The board runs standard versions of FLEX(tm) sold for 5" drives on 5" drives without modification. 8" versions sold at the time of this writing will require modification. We hope that this will change when a "Universal" FLEX(tm) is sold by TSC soon. There are also other software packages sold that will run on this board. One important point, some software sold for the 1771 chip can not be read by the 1791. This is because of the way the diskettes are initialized. The original "MINIFLEX" sold by SWTPC is one of these. Also some versions of FLEX 2.0. FLEX 9.0 for the 6809 can be read by either the 1771 or the 1791. FLEX 9.0 also has provisions for double density. Unfortunately, at this writing, we do not have the information necessary to know how to use the double density capability with the MDI-1. We hope to provide this information at a later date.

The MDI-1 has a data separator for use with the 1791. The 1771 must use its built-in separator or use a separator external to the MDI-1. There are several jumper options that set up for configurations using either FDC chip. These will be covered by describing what each jumper does and giving examples of configurations. The jumper descriptions follow:

J1 - Connects the address decoder to the option select register so that the register is addressed at CS5 address + 1. (Examples: 8015 for 6800 systems, I015 for 6809 systems.)

J2 - Connects a gate combining TR00 signal and Side 0 signal to the density select line allowing automatic switching to single density for side 0 and track 00 combinations. This can be useful in some systems for IBM compatibility.

J3 - Connects part of the on-board separator to FDC pin 27. This jumper is used when a 1791 FDC is installed.

J4 - Connects unseparated or separated data to FDC pin 27. This jumper is used when a 1771 chip is installed.

J5 - Connects unseparated data to a 1771 when the 1771 internal separator is used with an 8" drive.

J6 - Connects separated data to a 1771 when an external separator is used with an 8" drive.

J7 - Connects part of the on-board separator to FDC pin 26 when a 1791 chip is used.

## MDI-1 USERS MANUAL

J8 - Connects a separated clock to FDC pin 26 when an external separator is used with a 1771 chip and an 8" drive.

J9 - Connects minus 5 volts to the FDC when a 1771 is used. Never install this jumper when a 1791 is used.

J10 - Connects the double density/single density select signal to the FDC when a 1791 is used.

J11 - Should be installed only when an external separator is to be used with a 1771 chip.

J12 - Connects a "READY" signal from an 8" drive to the FDC. This jumper should only be installed if the software checks for "READY".

There are also several connection points on the board that could be useful in some applications. See the schematics for locations.

### Example configurations:

To use the MDI-1 in a 6800 system with a 1771 and standard software such as FLEX(tm), J1, J4 and J9 should be installed. This assumes a 5" drive(s). If you wish to add 8" drive(s), you should install J5 when using the built-in separator of the 1771. Note that using the built-in separator with an 8" drive is frowned upon by some "authorities". We have found that it works reliably enough in non-critical applications. Using the built-in separator affects only reading, not writing, and most software will automatically re-try when the separator is occasionally unable to correctly read a byte. Note also that when using an 8" drive, you must run the processor at 1.25 mhz. This is slow enough that most standard 1mhz parts will operate reliably. When using the CPU-2, you can install a 5mhz crystal. When using the MP-A2, a resistor can be selected and bridged across the timing resistor for the 6875. When using the AD209 adapter board with the MP-A2, use a 5mhz crystal. In these applications, the MDI-1 is usually installed in slot 6. A jumper must be installed between the two end pins of slot 5 to jumper CS5 to the user defined line so that it is available at slot 6.

To use the MDI-1 with a 1791, install J1, J3, J7 and J10(install J10 only if you expect to use double density). Install J5 if you will also be using 8" drive(s). DO NOT install J9. Using double density on the 5" drive requires 1.25 mhz operation. See notes above. The notes above about slot 5 and 6 also apply to most applications. Remember that the 1791 can not read some earlier software. Make sure your software is compatible.

### Assembly:

Before beginning assembly, study the board description to determine which parts you will be installing. We recommend using a socket for the FDC chip, others are optional. Refer to the schematics, the parts placement diagrams, and the parts list in the following steps. Use a soldering iron with a fine tip, keep it CLEAN,

and use solder of a size suitable for the size of the pads you are working on.

1. Check the board carefully for possible foil breaks and solder bridges. The board is inspected by the manufacturer, so it is unlikely you will find any, but it is possible. A break or a bridge could be difficult to find and repair later, especially on the front side where components could cover it.
2. Install all the resistors on the board, space the 1/2 watt resistor a little above the board to dissipate heat. R13 is nominally 10k but may have to be selected in some cases. The width of the pulse coming out at J3 should be 175 nanoseconds plus or minus 50 nanoseconds. If difficulty in reading with the 1791 is encountered, this signal should be checked. In the boards we have built up, 10k has been the right value to use.
3. Install all capacitors on the board. Watch polarity on the tantalum.
4. Install the IC regulator on its pad in the lower left hand corner. A heat sink is not necessary unless you run very high input voltage.
5. Install the three 10 pin molex connectors along the bottom edge. These connectors should be inserted from the front side and pressed down so they seat firmly. Solder only one pin on each connector and check for proper alignment. Solder the remaining pins. Insert a molex index pin into the connector at the position of the index pin on the mother board.
6. Install the header(s) at the top edge. Insert from the front side.
7. Install the crystal. There are 2 holes for strapping down the crystal with bare wire if desired.
8. Install the Zener diode(s). The 5 volt one is not required if using the 1791. Watch polarity. 9. Install the jumpers for your application.
10. Now would be a good time to power up the board and check the output of the regulator and the zener(s). This will let you know if there are any power shorts so far and check the regulator and diode(s).
11. Install socket(s). Observe any orientation markings. Pin 1 is "up" on IC1 & IC2 and points to the right edge on all others.
12. Begin installing the ICs. We recommend you install several then power up and check voltages. This could help in case a bridge occurs. You'll know approximately where. Continue installing and checking until all except IC3 are installed. Leave it out until all is finished and observe static precautions when installing it.
13. This concludes assembly. Recheck your work looking for missed parts, solder bridges, misorientation of parts, poor solder joints, etc.

## MDI-1 Parts list:

### Resistors:

R1-R3, R14,R15	150 ohm 1/4 watt 5%
R4	33k ohm 1/4 watt 5%
R5	1 megohm 1/4 watt 5%
R6	4.7 megohm 1/4 watt 5%
R7-R9	1000 ohm 1/4 watt 5%
R10	5600 ohm 1/4 watt 5%
R11,R12,R26	510 ohm 1/4 watt 5%
R13	10k ohm 1/4 watt 5% (approx. see text)
R16-R19, R22,R25	10k ohm 1/4 watt 5%
R20,R21	4.7k ohm 1/4 watt 5%
R27	68 ohm 1/2 watt 5%

### Capacitors:

C1,C3,C4	.01 mfd. disc ceramic
C2, C7-C12	.1 mfd disc ceramic (bypass type)
C5	4.7 mfd 16v. tantalum
C6	22 pf silver mica or polystyrene

### Integrated circuits:

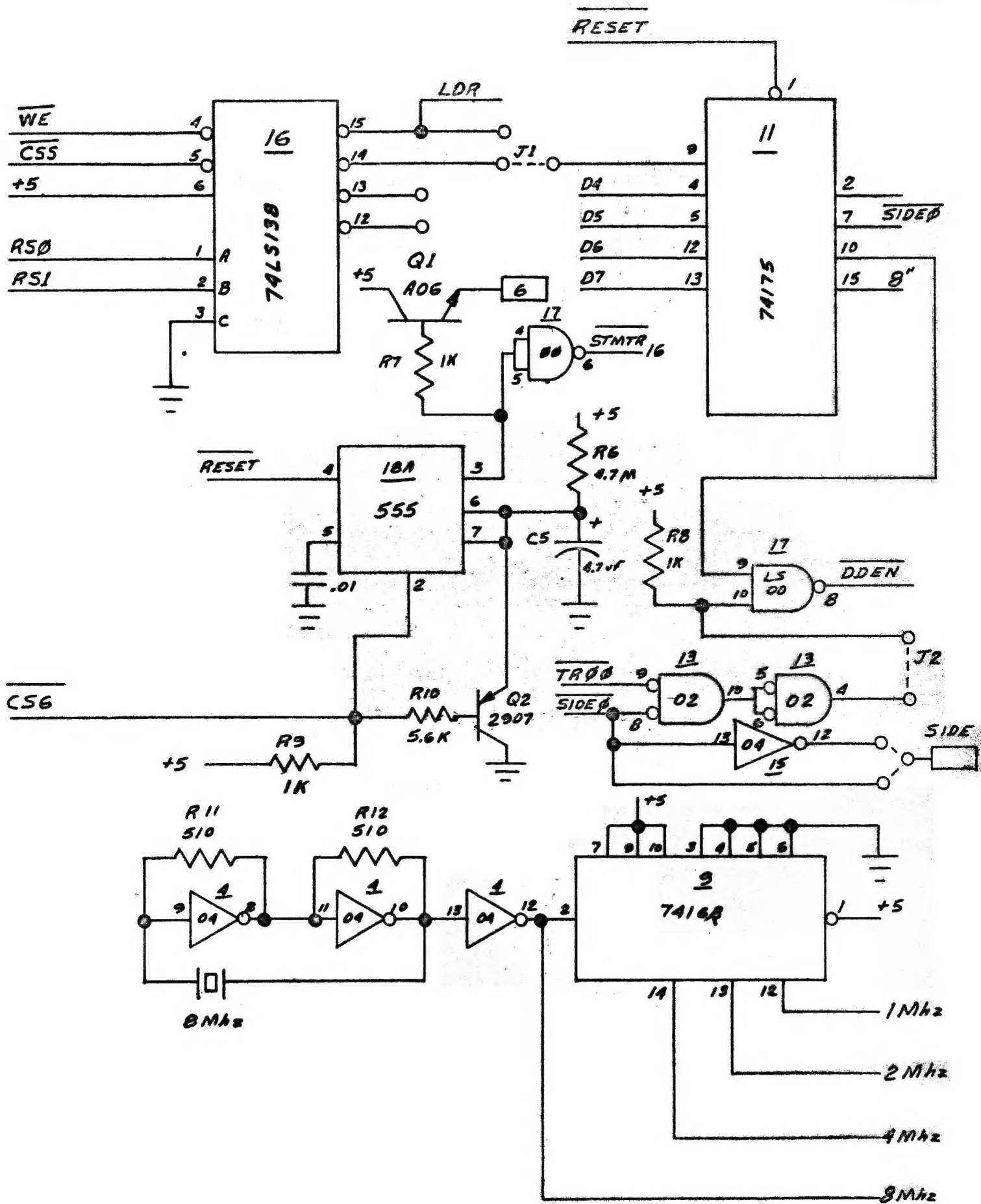
IC1,IC2	DS8835 transceiver
IC3	1771 or 1791 FDC according to application
IC4, IC15	7404 inverter
IC5	7432 or gates
IC6,IC11	74175 quad latches
IC7	7406 driver
IC8	74367 buffer
IC9	74163 counter
IC10	74193 counter
IC12	7442 decoder
IC13	7402 nor gate
IC14	7451 and or invert gate
IC16	74LS138 decoder
IC17	7400 nand gate
IC18A,IC18B	555 timer
IC19	74LS123 timer
IC20	74LS74 flip flops

### Transistors and Diodes:

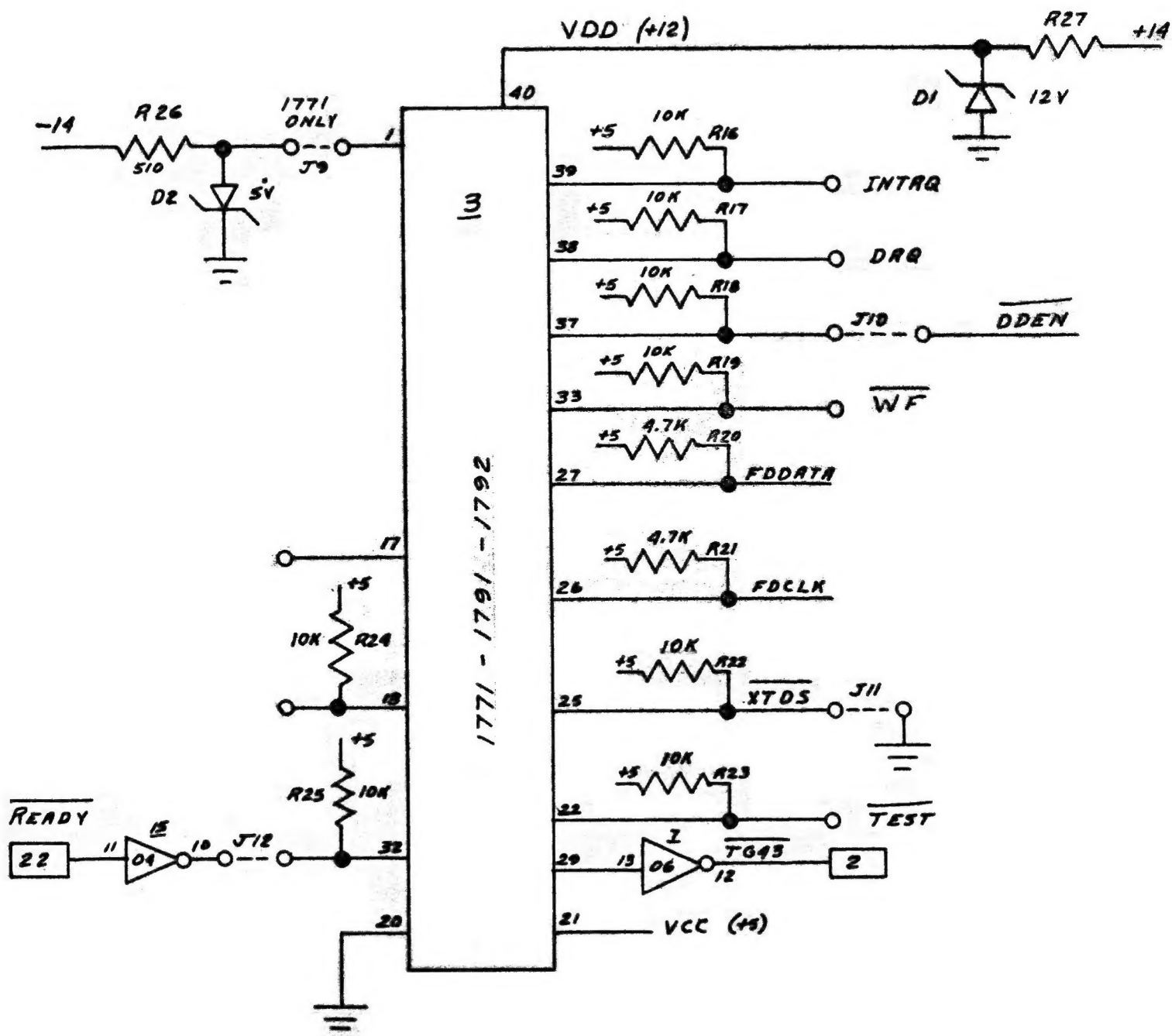
Q1	MPSA06 npn or equiv
Q2	2n2907 pnp or equiv
D1	12v. 1 watt zener
D2	5v. 1 watt zener

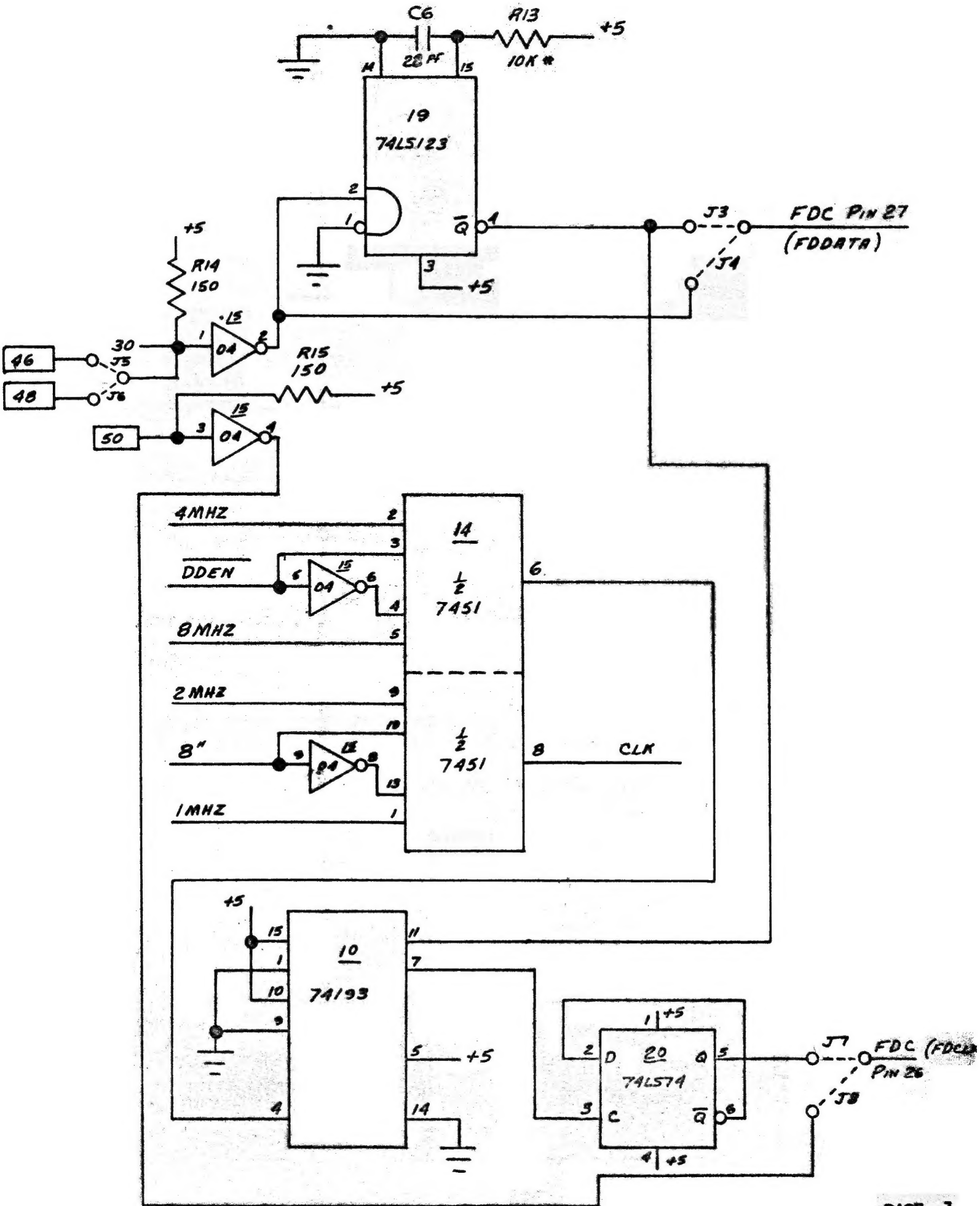
### Misc:

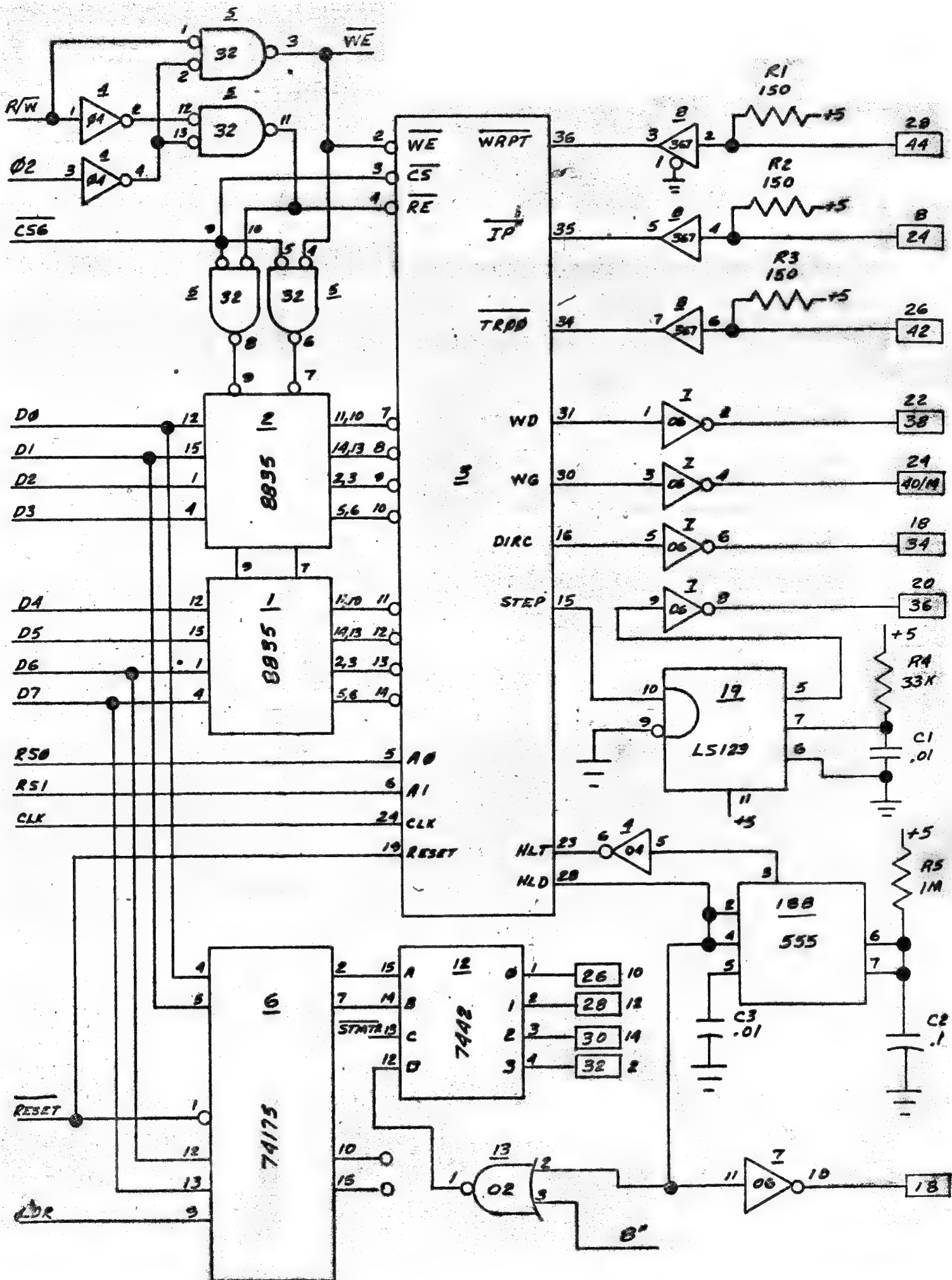
3	10 pin female molex #09-52-3101
1	34 pin male ribbon cable header
1	50 pin male ribbon cable header
1	8mhz crystal
1	7805 5volt regulator

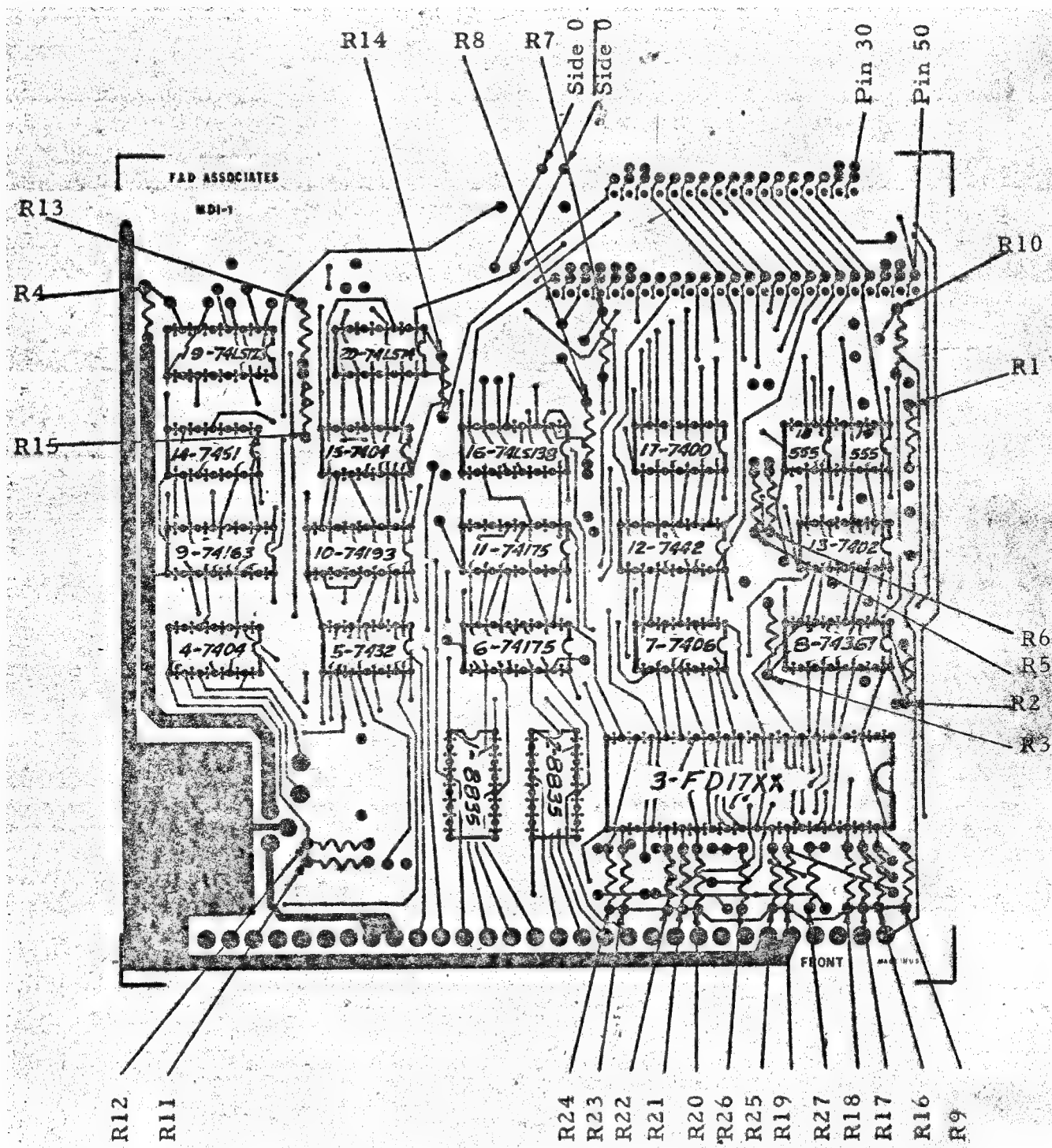


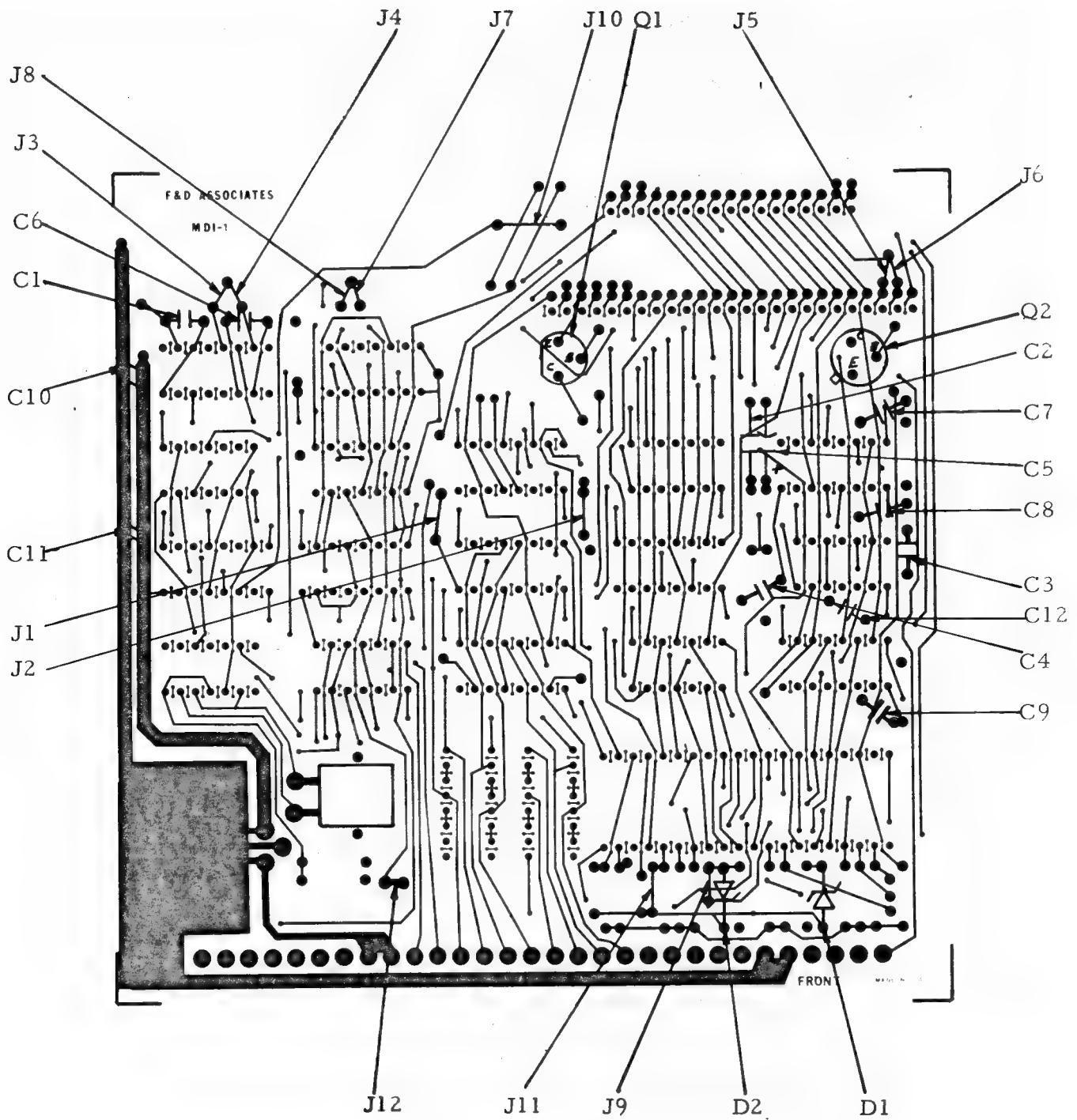












01/28/80

Addendum to MDI-1 Documentation:

After the MDI-1 initial batch of boards was received, we discovered that we needed to be able to separately switch the data separator clock frequency and the density select line in some instances. (8" drive using 1791 and on-board separator or 8" drive using 1771 and external data separator. To implement this change do the following:

On the back side of the board there are two foils connected to IC15 pin 5. One goes to IC14 pin 3 and the other leads away to IC17 pin 8. Cut the one leading to IC17 pin 8.

In the approximate middle of the foil between IC15 pin 5 and IC14 pin 3, there is a thru hole that connects a foil on the top side of the board. This foil goes to one pad for jumper J10. Cut this foil. If J10 is to be used (density switching), jumper from the uncut side of J10 area to a thru hole in the foil leading to IC17 pin 8. If jumper J10 was not to be used in your application, ignore this connection.

Solder a jumper from IC15 pin 5 to IC17 pin 3. Solder a jumper from IC17 pins 1 and 2 to IC11 pin 2.

This whole mod can be ignored if you only plan to use 5" or will use the internal separator of a 1771 to control an 8" drive.

-----

With our original design, running the CPU at 1.25mhz. puts the 1771 out of spec. regarding the read cycle. We have found one chip, not made by Western Digital, that could not keep up. Making a simple change in the read enable logic corrects the out-of-spec. condition.

Cut the foil from IC3 pin 4 to IC5 pins 10 and 11. Cut near IC3 so as not to disturb the connection from 10 to 11 on IC5.

Jumper IC3 pin 4 to IC4 pin 2.

This mod is advisable if you will be using 8" drives. It is unnecessary for the 5" unless using a speeded-up CPU for some reason. Running higher than 1.25 mhz is not recommended.

-----

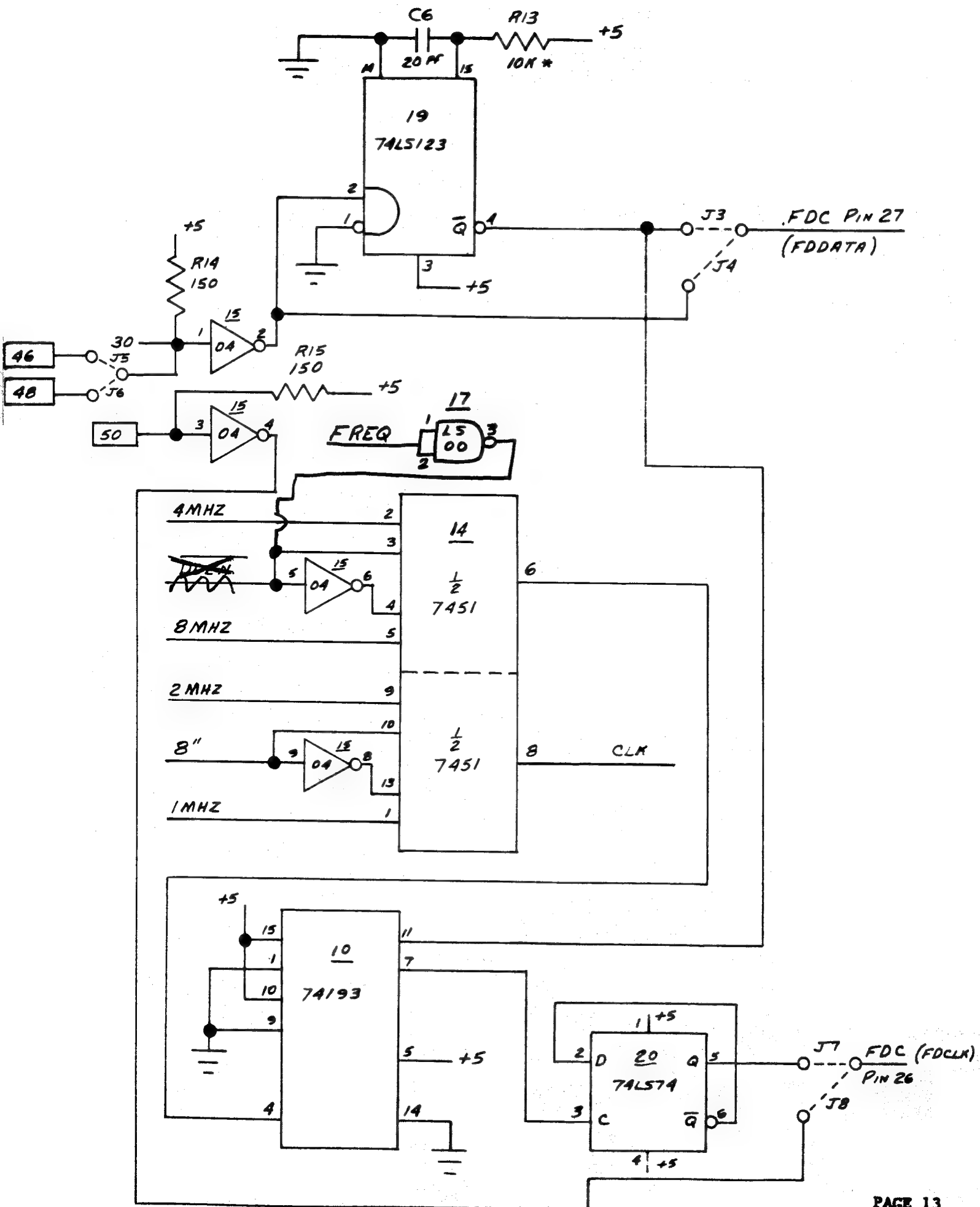
Both the above mods will be implemented in future production runs of the MDI-1..

02/20/80

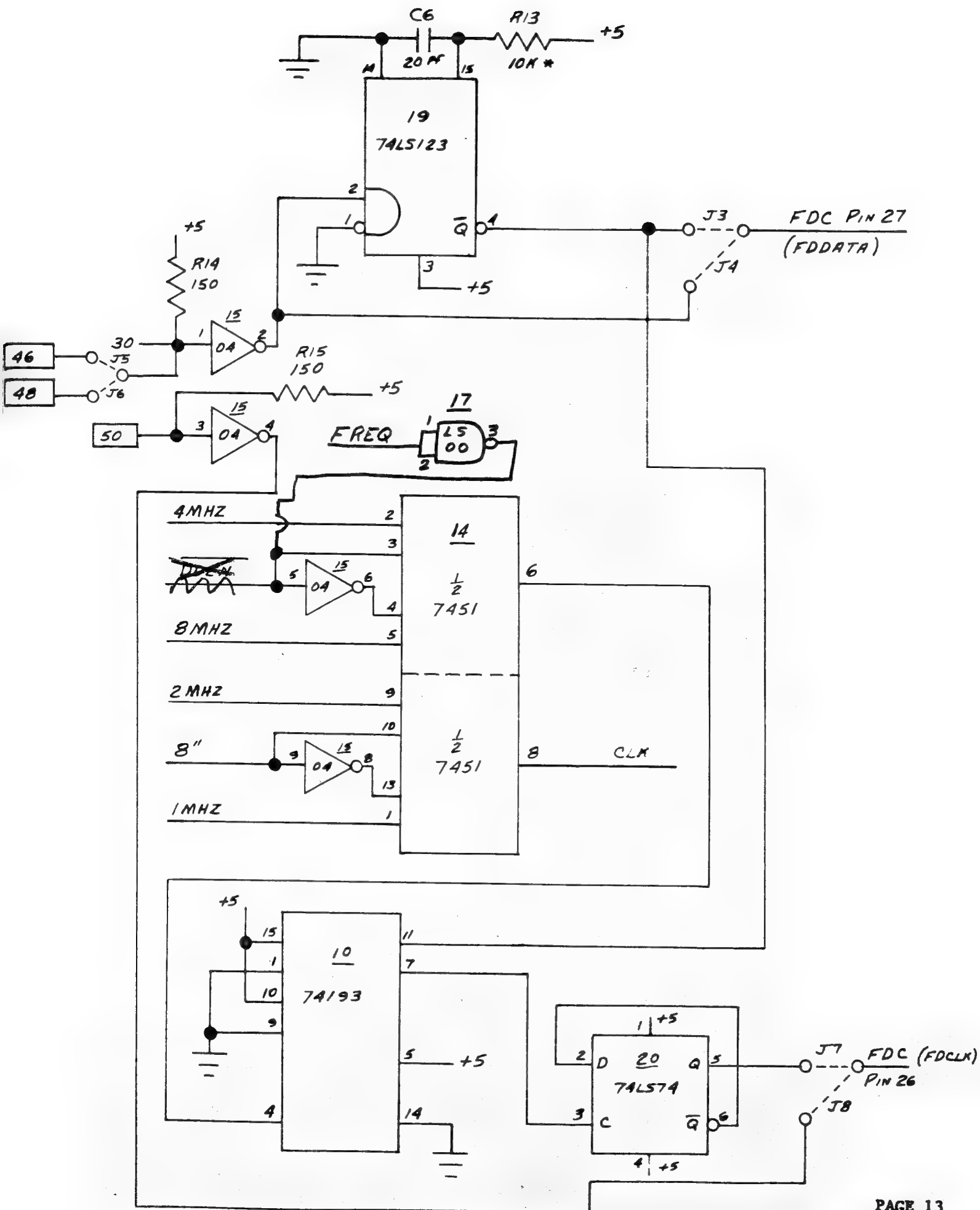
MDI-1 fix for bad NEWDISK

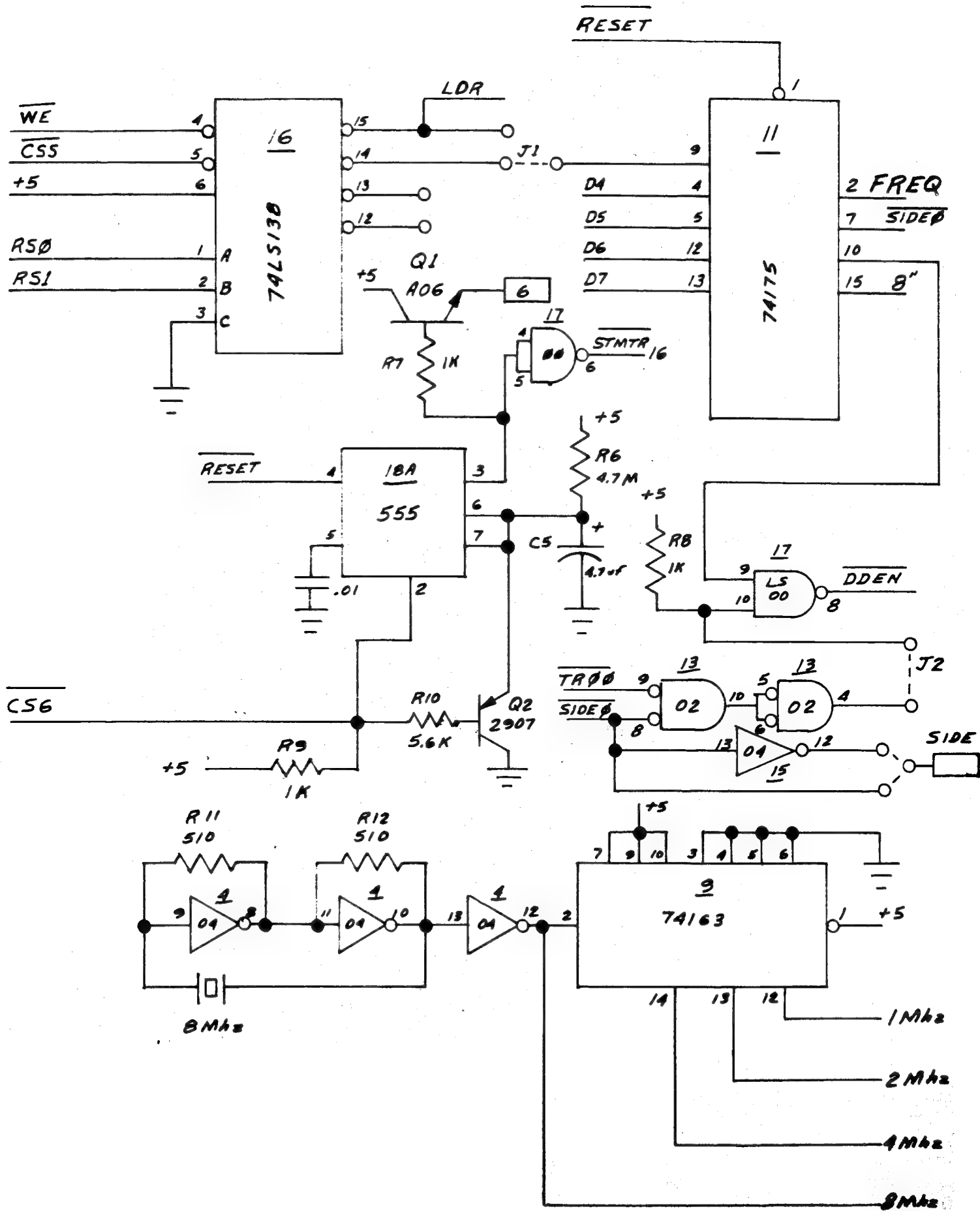
We have just been informed by one customer of many FATAL ERROR messages when using NEWDISK under FLEX 2.0. When there was no FATAL ERROR, he was getting many BAD SECTOR errors. We had not experienced this before but the fix was simple: a 100pf disc capacitor from IC7 pin one to ground. You can drill a small hole near IC7 pin one and use a nearby hole to pick up the ground or just solder the cap on the back side directly to pin one (IC7) and pin 7 which is ground (also IC7).

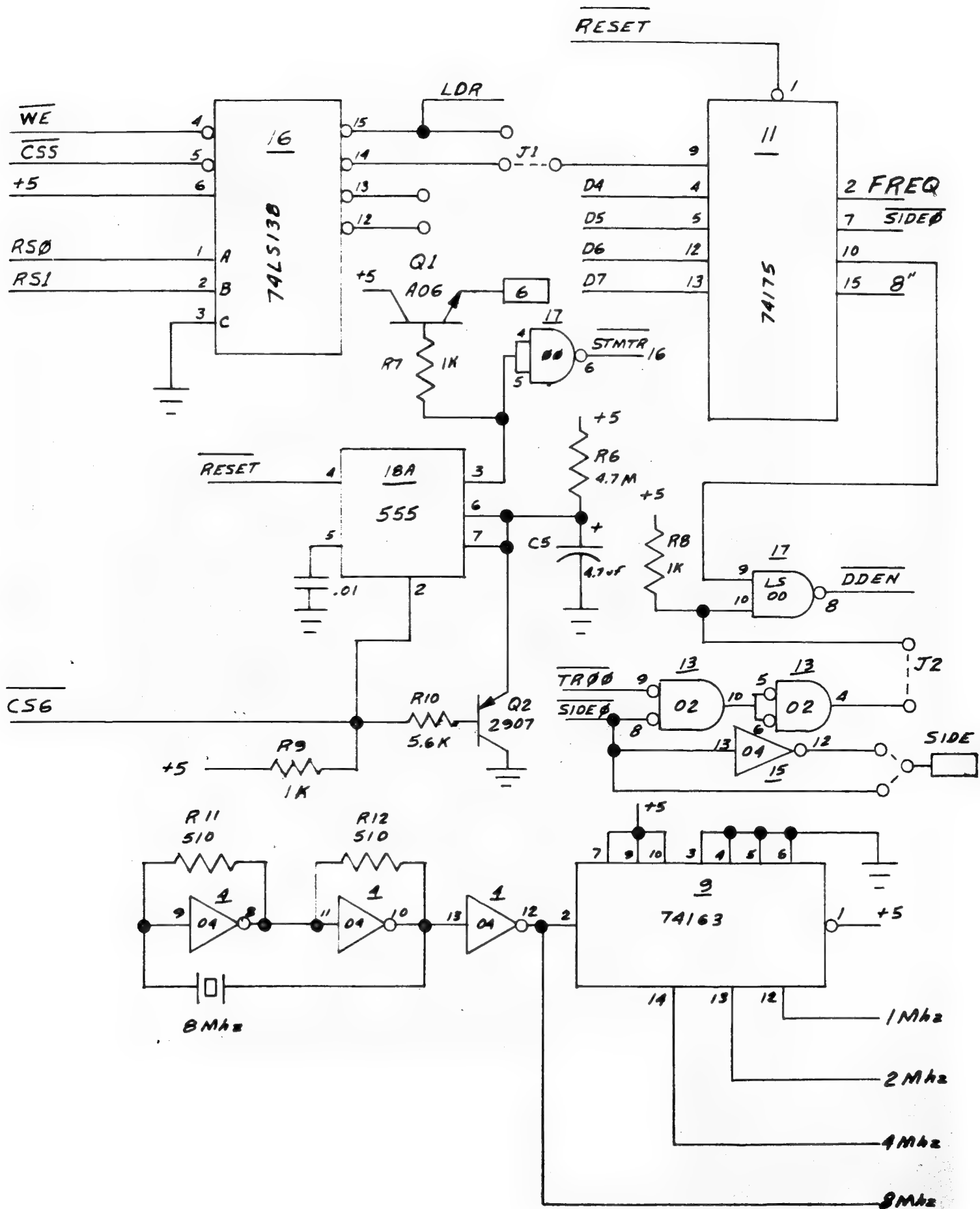
This problem is apparently due to too sharp a rise time on the WRITE DATA pulse which causes ringing on the cable to the drive. We have not experienced the problem but it could show up in some cases due to the particular cable length, system noise, etc. In any case, its probably best to go ahead and install the capacitor. We plan to add it to future production runs of the board.

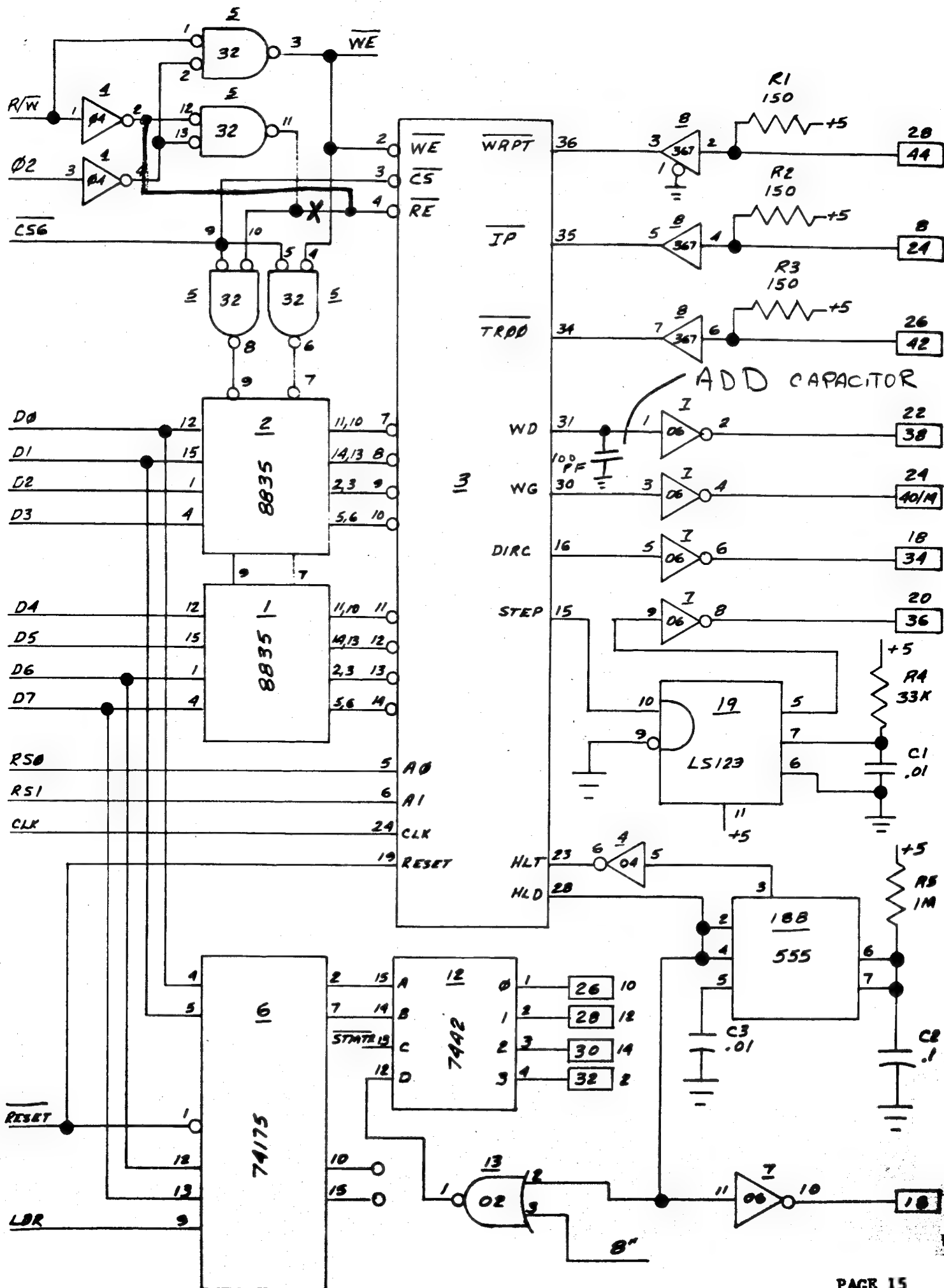


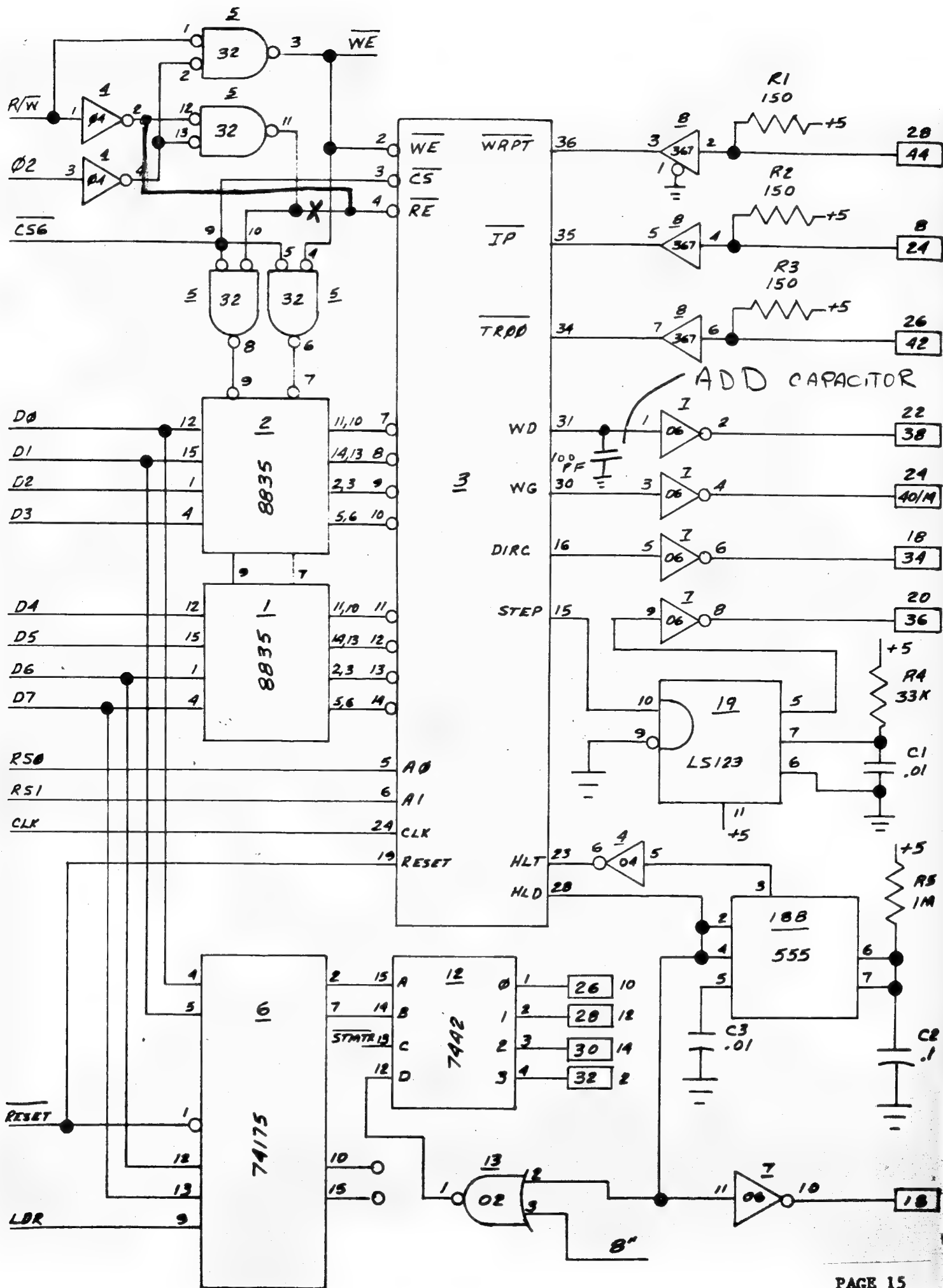












## NOTES ON USING MDI-1 WITH 8" DRIVES

Your system must be run at 1.25 mhz. to use 8" drives with the MDI-1. So far we have found no standard 1mhz. memory, CPU, ACIA, PIA, etc., unable to run at 1.25mhz. We have not, as yet, tested any dynamic memory and do not know for sure about it. However, we think that it will work unless very slow chips are used.

We have been able to test several 8" drives and find that a common problem is that they go too far outward when doing a RESTORE operation. This is usually due to the mechanical stop being incorrectly set. Our bootstrap allows for this by first doing a RESTORE, stepping in a few steps, and then doing another RESTORE.

The most critical operation in using the 8" drive seems to be in booting it up. This is due in part to failure to position over track 00 as mentioned above and to the failure of most bootstrap loaders to re-try in case a sector is loaded with errors. Our bootstrap program also includes extensive error checking and re-try capability to minimize this problem. This is the reason for its extra length as opposed to the simple ones used for 5" drives.

We have had good results with the 1771 and both 5 and 8 inch drives using the built-in separator in the 1771. However, this has to be somewhat dependent on the quality of the drive, quality of the diskettes, and whether or not the drive doing the reading wrote the diskette in the first place.(interchangeability) In case some people run into difficulty with using the internal separator and do not have a drive that has a suitable built-in separator of its own, we have decided to offer an external separator as a plug in option to the MDI-1. This could also be useful if you have a system that is marginal and is running excessively slow due to having to do a lot of re-tries. This board will be available soon. In the meantime, if anyone wants to wire-wrap it, we will be glad to supply the schematic. Actually the interconnections are few enough that you could construct it with point-to-point wiring on perf board.

A symptom of excessive re-trying on reads is a large number of bad sectors during initialization of a diskette that do not occur in the same spot if you do the initialization over.

We have seen some drives that will interchange diskettes except for track 00 (they won't boot). This is an indication that the track 00 switch or positive stop is mis-located. The drive should be re-aligned.

NAM BOOT  
OPT PAG

\*  
\*A BOOTSTRAP PROGRAM FOR THE MDI-1  
\*WITH IMPROVED RESTORE AND RETRY FEATURES  
\*  
\*ASSEMBLED FOR EIGHT INCH DRIVE  
\*  
\*EXPECTS DISK TO BE BOOTED FROM DRIVE 0  
\*  
\*CAN BE CHANGED TO FIVE INCH AND/OR DRIVE 1 TO 3  
\*

8014 DRVREG EQU \$8014  
8015 CONREG EQU \$8015  
8018 CONREG EQU \$8018  
801A SECRES EQU \$801A  
801B DATREG EQU \$801B

0000 DRIVE EQU 00 CAN BE 01, 02, 03  
0090 SIZE EQU \$90 FOR EIGHT INCH, FIVE INCH IS 00

\*  
0100 ORG \$0100  
0100 86 00 ENTER LDA A #DRIVE  
0102 B7 80 14 STA A DRVREG  
0105 86 90 LDA A #SIZE  
0107 B7 80 15 STA A CONREG  
010A 8D 60 BSR WAITB WAIT FOR THE BUSY FLAG TO CLEAR  
010C 8D 66 BSR RESTOR  
010E 8D 51 BSR STEPIN STEP IN, IN CASE HEAD IS BEHIND SWITCH  
0110 8D 62 BSR RESTOR  
0112 CE FF FF LDX #FFFF  
0115 08 WAITM INX WAIT FOR MTR UP TO SPEED  
0116 09 DEX  
0117 09 DEX  
0118 26 FB BNE WAITM  
011A 7F 80 1A CLR SECRES  
011D 8D 3D BSR WAITDI  
011F CE 24 00 LDX #2400  
0122 8D 19 BSR READ1 GET TRACK 00 SECTOR 00  
0124 26 DA BNE ENTER BAD READ, START OVER  
0126 86 01 LDA A #1  
0128 B7 80 1A STA A SECRES  
012B 8D 2F BSR WAITDI  
012D 8D 0E BSR READ1 GET TRACK 00 SECTOR 01  
012F 26 CF BNE ENTER START OVER IF ERROR CODE  
0131 86 02 LDA A #2  
0133 B7 80 1A STA A SECRES  
0136 8D 05 BSR READ1 GET TRACK 00 SECTOR 02  
0138 26 C6 BNE ENTER START OVER IF BAD READ  
013A 7E 24 00 JMP \$2400 ENTER LOADER

\*READ SUBROUTINE

```

*
013D 86 8C      READ1  LDA A  #8C      READ SECTOR COMMAND
013F B7 80 18      STA A  COMREG
0142 8D 18          BSR  WAITDI
0144 F6 80 18  READ2  LDA B  COMREG
0147 C5 02          BIT B  #2
0149 26 06          BNE  READ3
014B C5 01          BIT B  #1
014D 26 F5          BNE  READ2
014F 20 08          BRA  SECRED
0151 B6 80 1B  READ3  LDA A  DATREG
0154 A7 00          STA A  0,X
0156 08            INX
0157 20 EB          BRA  READ2
0159 C4 1C      SECRED AND B  #1C
015B 39            RTS

```

```

*
*SUBROUTINES
*

```

```

015C 8D 00      WAITDI BSR  DUM1
015E 8D 00      DUM1  BSR  DUM2
0160 39          DUM2  RTS
0161 8D 02      STEPIN BSR  STEPI2
0163 8D 00          BSR  STEPI2
0165 86 5B      STEPI2 LDA A  #5B      STEP IN COMMAND
0167 B7 80 18      STA A  COMREG
016A 8D F0          BSR  WAITDI
016C F6 80 1B  WAITB  LDA B  COMREG
016F C5 01          BIT B  #1
0171 26 F9          BNE  WAITB
0173 39            RTS
0174 86 0B      RESTOR LDA A  #0B
0176 B7 80 18      STA A  COMREG
0179 8D E1          BSR  WAITDI
017B 8D EF          BSR  WAITB
017D 39            RTS
END      ENTER

```

NO ERROR(S) DETECTED

+++



The following instruction sheets are sent with the MDI instructions so that you can see what is involved in adding 8" drives and using FLEX(tm) software. The example provided is for modifying 5" FLEX 2.0. The other versions follow a similar pattern. The 5" versions are supplied on a 5" diskette and the 8" versions are supplied on 8" diskettes.

The price of each version is \$20.00.

We currently have available versions for 5" FLEX 2.0, 8" FLEX 1.0 for SWTPC (the one supplied with the original single density drives) and 8" FLEX 1.0 for SMOKE SIGNAL. When ordering, be sure to state which version.

## INSTRUCTIONS FOR MAKING FLEX 2.0 COMMUNICATE WITH 8" DRIVE

The supplied diskette contains several files important to converting to a mixed 5 inch and 8 inch system. They are: MDIDRV.BIN- the new disk drivers; INIT.CMD - a new formatter that formats both 5" and 8" diskettes; SETDRV.CMD - command utility to assign drive sizes. A complete description of these programs will be given later.

Also a new LOADER program has been provided on track zero of the supplied diskette. This loader is very necessary to the conversion process.

Now for the conversion steps:

1. Make sure that your CPU board has been set up to run at 1.25mhz.
2. Load the supplied diskette into drive 0 and type "D" to boot it up. If your monitor doesn't contain a bootstrap, you will have to load one into RAM and execute it. If the boot operation fails, push reset and wait about 10 seconds before trying again. The reason for waiting is that the controller chip will be "busy" and most bootstrap programs do not check for this signal. Booting again right away will almost always fail. When the operation is successful, the loader will prompt with a "pound" sign.
3. Remove the supplied diskette and put in a FLEX 2.0(tm) system diskette that has FLEX(tm) and has been linked.
4. Type "B". The loader should search out FLEX(tm) and load it in. "Starting Address is AD00" should then be printed on the terminal (by the loader program). The original code for FLEX(tm) is now in memory.
5. Again switch the diskettes, putting the supplied diskette back into drive 0.
6. Type "B" again. This time the new drivers will be loaded over the original FLEX(tm). "Starting Address is AD00" should be output to the terminal once more.
7. If no patches have to be made to suit your system, simply type "G" and FLEX(tm) should come up and prompt you for the date. If you need to make some patches, push reset, make the patches, and start the program at AD00. An example of a patch that might need to be made now is when your system will have drive zero defined as an eight inch drive. In this case, you should push reset and use the monitor to change location \$BEB3 to \$80. (The drivers, as they are supplied, set up to recognize drive 0 as a five inch, drive 1 as an eight inch and drives 2 and 3 as five inch.) Also change the following three locations to suit your system if necessary. See the description of MDIDRV.BIN. Since the assignments for the monitor I/O is in this area, you can also put in custom locations at this time. See "ADAPTING FLEX TO CUSTOM MONITORS" in the FLEX manual for more information.
8. You can use this procedure for bringing up FLEX(tm) each time or you can use normal FLEX(tm) procedures to "APPEND,FLEX.COR,MDIDRV.BIN,FLEXM.SYS". This gives you a new file to be linked and booted. If you made patches before starting FLEX(tm), you should first "SAVE,MDIDRV.BIN,BE80,BFFF,AD00" and then you should "APPEND,FLEX.COR,MDIDRV.BIN,FLEXM.SYS". Now, if you will "LINK,FLEXM.SYS", you can boot up the system with the changes already made.

## UTILITY DESCRIPTIONS FOR 5" TO 8" CONVERSION

### MDIDRV.BIN

This is a binary file containing the new drivers that will overlay the original. The only locations that should be of any interest to the user are the ones that contain the drive assignments for the four possible drives. These locations are the four bytes beginning at \$BEB3 which contains the status of drive 0. The others are in ascending order. The status byte should contain \$00 if the drive is 5" and \$80 if it is an 8". These locations are normally set by the "SETDRV" command but may be patched manually if desired. These bytes will also later be used to reflect "single density, double density" status.

### SETDRV.CMD

This is a utility for setting the drive status. When called, it prints the current status of each drive and allows a change if desired. Responding with a single digit (5 or 8) will set the status of that drive. Responding with a carriage return will skip to the next drive with no change. Responding with a "D" for "done" will cause an exit back to FLEX(tm). Responding with anything else will cause a "?" and a new prompt to be printed.

### INIT.CMD

This is the diskette formatter. When called, it asks for the drive which should be a single digit from 0 to 3. It then asks for the size which should be 5 or 8. Next, it prompts for date which should be in the form MM/DD/YY. The slashes are printed automatically. Now it asks "SCRATCH DISK LOADED?" and if you answer with a "Y", the formatting is begun. If bad sectors are found, error messages will be printed and they will be removed from the free chain. Certain sectors must be good or the formatting will be aborted and an appropriate message printed. These include: all of track 00; track 01, sector 01; and the last sector of the last track. When the formatting is done, the program loops back to the beginning. If no more diskettes are to be initialized at this time, type "X" to exit back to FLEX(tm).

### LOADER

This program is written onto track 00 sectors 00-02 during the formatting process. It is brought into memory by the bootstrap loader and prompts with a "pound" sign. This program contains 3 commands:

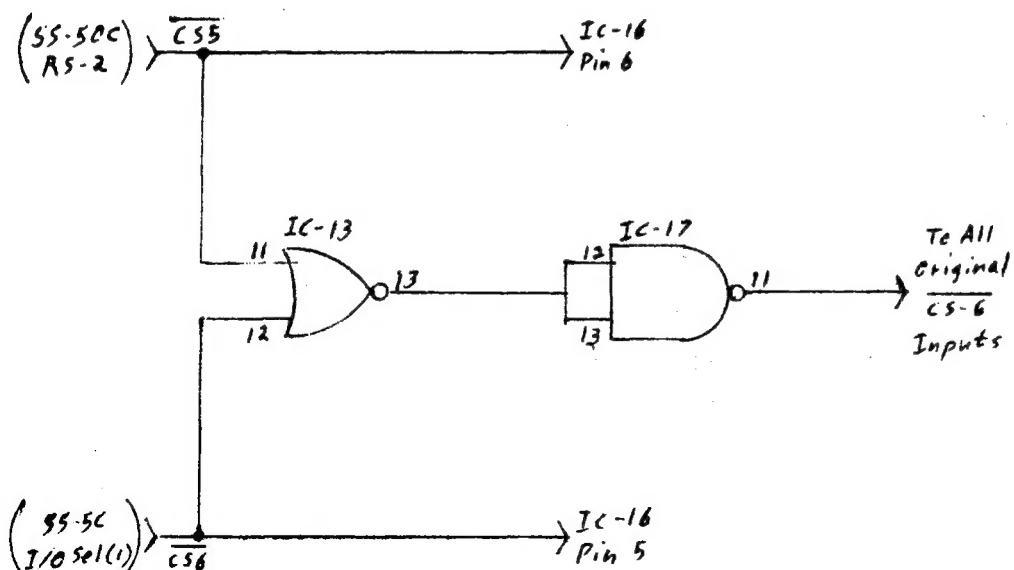
B -BOOT- Loads the program that has been "linked" into memory and prints out its starting address.

L -LOAD- Loads in the file beginning at a specified track and sector and prints out its starting address.

G -GO- Jumps to the starting address printed out by one of the above commands

Note: All these operations assume drive 0. Also, when booting from an 8" drive using one of the standard bootstraps, you should first store \$80 in \$8015 to tell the MDI-1 that it will be using an 8" drive. RESET always returns the controller to the 5" mode.

# MODIFICATION TO MDI-1 BOARD FOR S-50C BUS



## NOTES:

1. S-50C BUS 30 PIN I/O DESIGNATIONS IN PARENTHESIS.
2. THREE FOIL CUTS AS FOLLOWS:
  - A. +5V TO IC-16, PIN 6.
  - B.  $\overline{CS5}$  TO IC-16, PIN 5.
  - C.  $\overline{CS6}$  NEAR THE POINT WHERE IT COMES ON THE BOARD (BEFORE IT BRANCHES TO IT'S VARIOUS CONNECTING POINTS).
3. MDI-1 BOARD INSTALLED IN SLOT #1 OF S-50C BUS DECODED TO E01X.
4. NO SOFTWARE CHANGES REQUIRED IN STANDARD TSC FLEX 9.0 FOR SWTPC COMPUTERS.